

public static methods on an interface (as of JDK8)

Another enhancement that Java included in JDK 8, was support for public static methods on the interface.

Static methods don't need to specify a public access modifier, because it's implied.

When you call a public static method on an interface, you must use the interface name as a qualifier.

In the ArrayList lectures, you may remember I used two static helper methods, on the Comparator interface, which were added in JDK 8.

These were `Comparator.naturalOrder()` and `Comparator.reverseOrder()`.

Private methods (JDK 9)

JDK 9 gave us private methods, both static and not static.

This enhancement primarily addresses the problem of re-use of code, within concrete methods on an interface.

A private static method can be accessed by either a public static method, a default method, or a private non-static method.

A private non-static method is used to support default methods, and other private methods.